International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

Machine Learning-Based Prediction of Magnetic Properties from Hysteresis Curves: A Comparative Study of Random Forest, Gradient Boosting, Xgboost, Lightgbm and Support Vector Regressions

T. E. Awopejo[a], P. O. Adigun[b], N. A. A. Azeez[c], T. T. Oyekanmi[d], M. A. Adekanye[e], A. Obisesan[f]

Department of Natural Resources Management (Geology), New Mexico Highlands University, Las Vegas, New Mexico, USA [a]

Department of Computer Science, New Mexico Highlands University, Las Vegas, New Mexico, USA [bd]

Department of Physics, University of Abuja, Abuja, Federal Capital Territory, Nigeria [c]

Information Security Risk Manager, MIBA Group, Austria [e]

Laurea University of Applied Sciences, Vanta, Finland [f]

Email: tawopejo@live.nmhu.edu

Email: poadigun@nmhu.edu

Email: azeez.abimbola2019@uniabuja.edu.ng

Email: toyekanmi@live.nmhu.edu

Email: adekanye_michael@yahoo.com

Email: obisesan.abiodun@yahoo.com

ABSTRACT: This study investigates the application of machine learning regression models for predicting magnetic properties directly from hysteresis curves. A dataset of experimentally measured hysteresis loops was preprocessed and used to train five algorithms: Random Forest, Gradient Boosting, XGBoost, LightGBM, and Support Vector Regression (SVR). A dataset of 561 samples, each containing 800 resampled features and five target variables, was divided into training and testing sets to evaluate model performance using R^2, MAE, and RMSE metrics. Results revealed that Gradient Boosting and XGBoost consistently achieved superior accuracy, with Gradient Boosting excelling in intrinsic coercivity ($R^2 = 0.959$) and XGBoost dominating in remanence ($R^2 = 0.921$), coercivity, and residual induction. Support Vector Regression demonstrated specialized strength in saturation moment ($R^2 = 0.753$), though it underperformed in intrinsic coercivity. The study recommends prioritizing Gradient Boosting and XGBoost for multi-target regression tasks.

KEYWORDS: Magnetic properties, Hysteresis curves, Random Forest, Gradient Boosting, XGBoost, LightGBM, Support Vector Regression

I. INTRODUCTION

The main scientific approach to problems and challenges is to design a machine learning (ML) model that identifies the problem and proposes solutions to the problem entity, which, as a result of this process, develops knowledge and new technologies. The application of machine learning (ML) currently has vast coverage in many fields; namely, science, engineering, geology, geophysics, and medicine [1-6]. Machine learning applications have made many contributions to geology and geophysics [6-8].

Magnetic materials are important in modern technology, with applications ranging from energy storage and conversion to data storage, sensing, and biomedical devices [9, 10]. The performance of magnetic materials are majorly determined

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

by intrinsic magnetic properties such as coercivity, remanence, and saturation magnetization, which are typically derived from hysteresis curves [11-13]. A hysteresis curve provides a graphical representation of the relationship between magnetization and applied magnetic field, which captures the nonlinear and history-dependent behavior of magnetic materials [11, 14]. Accurate characterization of these curves is essential for material design, optimization, and deployment in industrial and scientific applications. Furthermore, the magnetic properties of a material are effectively described using a hysteresis curve, which shows the relationship between magnetic flux density (B) and magnetizing field strength (H). The curve indicates how a magnetic material responds when an external magnetic field is applied, increased, reduced, and eventually reversed.

Initially, as the magnetizing field increases, the magnetic flux density increases rapidly because the magnetic domains within the material progressively align with the applied field. With further increase in the field, the material approaches magnetic saturation (Bs), where most of the magnetic domains are aligned and further increases in the applied field produce only a small increase in magnetization. When the applied field is subsequently reduced to zero, the material does not return to its original unmagnetized state. Instead, some magnetism remains in the material; this residual magnetism is known as remanence or retentivity (Br) [15]. To remove this residual magnetism, a magnetic field must be applied in the opposite direction. The magnitude of the reverse field required to reduce the magnetization to zero is called the coercive field or coercivity (Hc) [14, 15]. These features of the hysteresis curve, saturation, retentivity, and coercivity, are important parameters for characterizing the magnetic behavior of materials [14].

Additionally, the shape and area of the hysteresis curve also provide important information about the practical magnetic properties of a material. A material with a narrow hysteresis loop generally has low coercivity and relatively low hysteresis energy loss, making it easy to magnetize and demagnetize. Such materials are referred to as soft magnetic materials and are suitable for transformer cores, electric motors, generators, relays, and electromagnets where the magnetic field changes repeatedly. In contrast, a wide hysteresis loop indicates high coercivity and high retentivity, meaning that the material strongly resists demagnetization and retains its magnetism after the external field is removed. These are known as hard magnetic materials and are commonly used for permanent magnets, magnetic recording devices, and other applications requiring persistent magnetization. In Figure 1, the area enclosed by the hysteresis loop represents the energy dissipated as heat per magnetization cycle, known as hysteresis loss. Therefore, the hysteresis curve provides a convenient way of determining and comparing important magnetic properties such as permeability, saturation, retentivity, coercivity, and energy loss, and it guides the selection of magnetic materials for specific engineering applications [14-17].

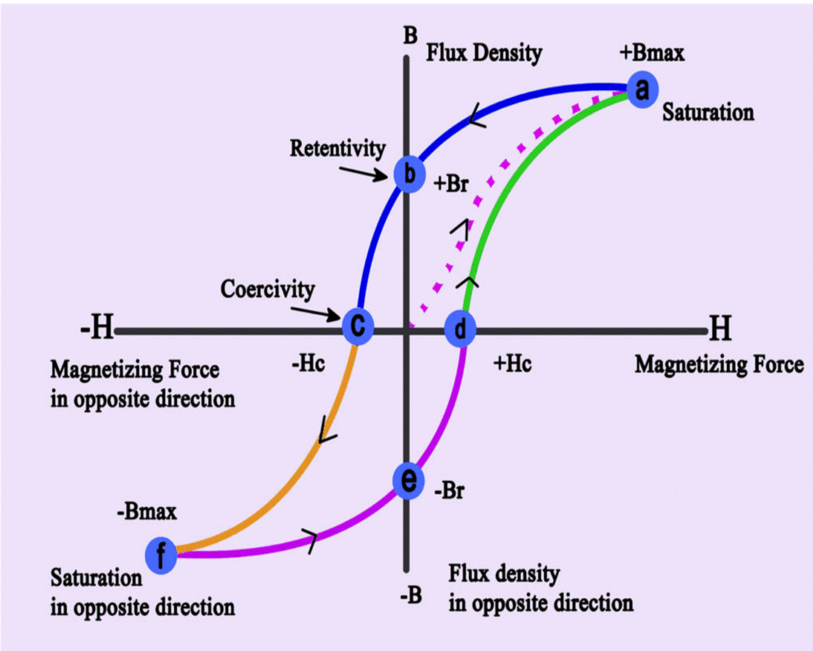


Figure 1: Hysteresis curve [16]

In recent years, machine learning (ML) has emerged as a transformative tool in materials science, offering data-driven solutions to predict, classify, and optimize material properties [1, 6]. Unlike conventional physics-based models, ML algorithms can learn complex nonlinear relationships directly from experimental data, which enables rapid predictions without requiring exhaustive theoretical derivations [18]. This paradigm shift has given rise to the field of materials informatics, where ML techniques are applied to accelerate discovery and characterization of advanced materials. Within this context, the prediction of magnetic properties from hysteresis curves represents an integration of geological, geophysical, and machine learning applications. The ML models are trained to capture the curve features and translate them into quantitative property estimates.

Several ML algorithms have demonstrated strong performance in regression tasks, particularly in domains where nonlinear dependencies dominate. Random Forest (RF), an ensemble method based on decision trees, is known for its robustness and interpretability [1, 19, 20]. Gradient Boosting (GB) improves predictive accuracy by sequentially correcting errors from prior models, while XGBoost and LightGBM represent optimized gradient boosting frameworks that offer scalability and efficiency for large datasets [15, 19, 21]. In contrast, Support Vector Regression (SVR) is built based on kernel functions to model nonlinear relationships, which provides strong theoretical guarantees but often struggles with scalability in high-dimensional datasets.

This study contributes to the growing body of work in materials informatics by conducting a comparative evaluation of five machine learning regression algorithms, Random Forest, Gradient Boosting, XGBoost, LightGBM, and Support Vector Regression (SVR), for predicting magnetic properties from hysteresis curves. The motivation for this research lies in bridging the gap between experimental magnetics research and computational prediction. While hysteresis measurements remain the gold standard for characterizing magnetic materials, they are limited by experimental constraints such as instrument precision, sample preparation, and environmental conditions. By integrating ML models, researchers can reduce reliance on repeated measurements, accelerate material screening, and potentially uncover hidden correlations between curve features and material performance. Moreover, predictive models can serve as complementary tools to guide experimental design, focusing resources on the most promising candidate materials.

Crucially, this research validates the potential of machine learning to accelerate materials characterization and design, which reduces experimental shortcomings and enables data-driven innovation. The integration of ML models can achieve high predictive accuracy from hysteresis curves and paves the way for broader adoption of computational methods in magnetics and materials science. Furthermore, the findings have implications for industrial applications, geological and seismological research, including the optimization of magnetic components in electronics, renewable energy systems, and medical devices, as well as for academic research seeking to advance the frontiers of materials informatics. Hence, a comparative study of these algorithms applied to hysteresis curve data is essential to identify the most effective approach for magnetic property prediction.

Therefore, this research aims to train and test a large dataset of rock-sample magnetic properties using machine learning; that is, this study will develop and evaluate machine learning regression models for predicting magnetic properties directly from hysteresis curves. The objectives of the research are:

- To collect, preprocess, and normalize hysteresis curve data to ensure consistency and suitability for machine learning modeling.
- To design and train regression models using Random Forest, Gradient Boosting, XGBoost, LightGBM, and SVR to predict key magnetic properties such as coercivity, remanence, residual induction, and saturation magnetization.
- To assess the predictive performance of each model using metrics including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R^2 score.

II. METHODS

2.1 Description of the Study

This study presents a design of a machine learning (ML) model trained to classify and interpret the magnetic properties based on the hysteresis curve. The model was programmed using supervised learning (SL), and five major machine learning algorithms (MLAs) were implemented. The machine learning algorithms implemented are Random Forest (RF), Gradient Boosting, XGBoost, LightGBM, and SVR. The dataset consists of experimentally measured hysteresis loops, preprocessed to extract relevant features and normalized for model training. Performance is assessed using standard regression metrics, including Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and R^2 score.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

2.2 Data Source and Characteristics

A primary dataset was used for the ML modeling. The dataset was sourced from the Laboratory of Geology, Department of Natural Resources Management (Geology), New Mexico Highlands University, Las Vegas, New Mexico, USA. The dataset was obtained and measured from rock samples. The dataset consisted of experimentally measured hysteresis curves representing the magnetic behavior of different rock samples.

The hysteresis measurement was carried out on sample 9H-N01N using an EM4-CSB magnet powered by an LS643 power supply. The measurement was performed with a 100% head amplitude, a Gap 3–SSVT setting, and an 86-LC coil set (serial number A13B2). The coil balance was 0.05. The magnetic moment measurement system was calibrated using a nickel (Ni) sphere with a standard ID of Model 730908. The calibration was performed at a field of 5000 Oe, with an expected moment of 6.92 emu, and the calibration value obtained was −79.6459 emu/V. The test sample had a volume of 1.00 cm^3, an area of 1.00 cm^2, and a mass of 0.163 g. A demagnetization factor of zero was used in the CGS system, indicating that no demagnetization correction was applied to the reported magnetic parameters.

The total number of samples was 577, with a minimum point of 106 and a maximum point of 1018. The median point was 814, and the mean point was 854. Each curve was digitized to capture the relationship between magnetization (M) and applied magnetic field (H). From these curves, key magnetic properties such as coercivity (Hc), remanence (Mr), residual induction, and saturation magnetization (Ms) were extracted as target variables for prediction.

2.3 Modeling Software

Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation[1, 22]. This programming language is used to code, model, train, and simulate the background radiation.

2.4 Machine Learning Models

Five regression algorithms were implemented and compared:

1. Random Forest (RF): is an ensemble of decision trees trained on bootstrap samples. Predictions were averaged across trees to reduce variance. Hyperparameters tuned included the number of trees, maximum depth, and minimum samples per split.
2. Gradient Boosting (GB): is a sequentially built weak learner where each new tree corrected the errors of the previous one. Learning rate and number of boosting stages were optimized to balance bias and variance.
3. XGBoost: is an optimized gradient boosting framework with regularization. Parameters such as maximum depth, subsampling rate, and learning rate were tuned using grid search. It is known for efficiency and scalability, making it suitable for large datasets.
4. LightGBM: is a gradient boosting framework designed for speed and memory efficiency. It utilizes histogram-based algorithms for faster training. Key hyperparameters include the number of leaves, learning rate, and boosting iterations.
5. Support Vector Regression (SVR): It models nonlinear relationships using kernel functions (RBF kernel). Tuned parameters such as C (regularization), epsilon (margin of tolerance), and gamma (kernel coefficient) . It provided a benchmark for kernel-based regression approaches.

2.5 Model Training and Validation

All models were implemented in Python using libraries such as numpy, pandas, matplotlib, Scikit-learn, XGBoost, LightGBM, and joblib. Training was conducted on a workstation equipped with GPU acceleration to handle computational demands. Hyperparameter tuning was performed using grid search with cross-validation, ensuring optimal model configurations.

2.6 Performance Evaluation

Model performance was assessed using standard regression metrics:

- Mean Absolute Error (MAE): Measures average magnitude of prediction errors.
- Root Mean Square Error (RMSE): Penalizes larger errors more heavily, providing insight into prediction stability.
- Coefficient of Determination (R^2): Indicates the proportion of variance in magnetic properties explained by the model.

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

2.7 Preprocessing

The dataset was preprocessed for training and testing. The following steps were initiated.

Figure 2: Procedural chart of data preprocessing of the magnetic properties dataset

Furthermore, the major manipulations of the dataset were:

- Normalization: Magnetization values (hysteresis curve lengths) were scaled to a uniform range from 814 original to 400 resampled points, to reduce bias from differing measurement magnitudes.
- Feature Extraction: 800 features, which comprise 400 resampled field values and 400 resampled moment values.
- Data Splitting: The dataset was divided into training (80%) and test (20%) sets.

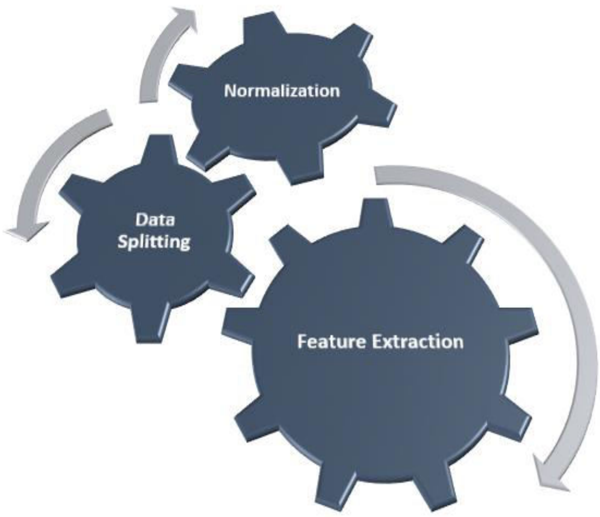


Figure 3: Data preprocessing of the dataset

The distribution of hysteresis curve lengths reveals important characteristics of the dataset used in this study. The majority of curves are concentrated around 800 measurement points, with over 400 samples falling into this category. A smaller

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

subset of approximately 130 curves contains 1,000 measurement points, while only a very limited number of curves are recorded at 100 measurement points. This clustering indicates that the dataset is dominated by medium-length hysteresis curves.

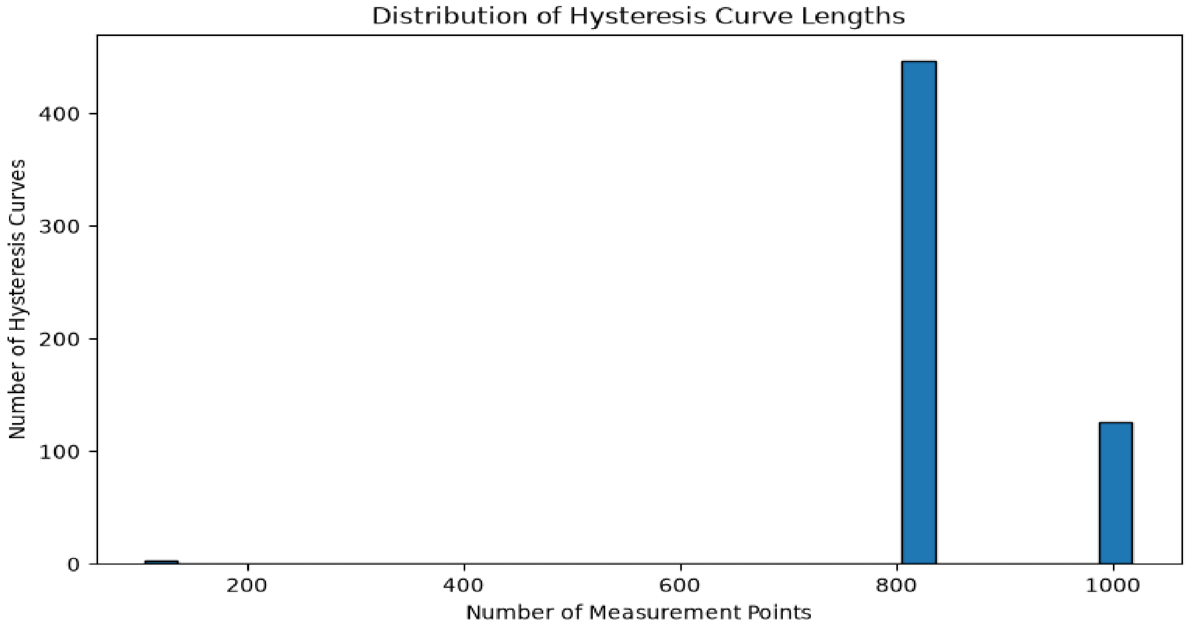


Figure 4: Visualization of the distribution of hysteresis curve lengths

```
TARGET_NAMES = [
    "Coercivity",
    "Intrinsic coercivity",
    "Remanence",
    "Saturation moment",
    "Residual induction"
]

TARGET_NAMES
```

```
['Coercivity',
 'Intrinsic coercivity',
 'Remanence',
 'Saturation moment',
 'Residual induction']
```

Snippet 1: Definition of prediction targets

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

```
original_field = complete_samples[0]["field"]
original_moment = complete_samples[0]["moment"]

resampled_field, resampled_moment = resample_curve(
    original_field,
    original_moment
)

print("Original points:", len(original_field))
print("Resampled points:", len(resampled_field))
```

```
Original points: 814
Resampled points: 400
```

Snippet 2: Test resampling of the data

The hysteresis curve labeled 18AK001 (Figure 5) illustrates the fundamental magnetic behavior of the material under study. The plot shows the relationship between the magnetic moment (M) and the applied magnetic field (H), displaying the characteristic loop associated with ferromagnetic materials.

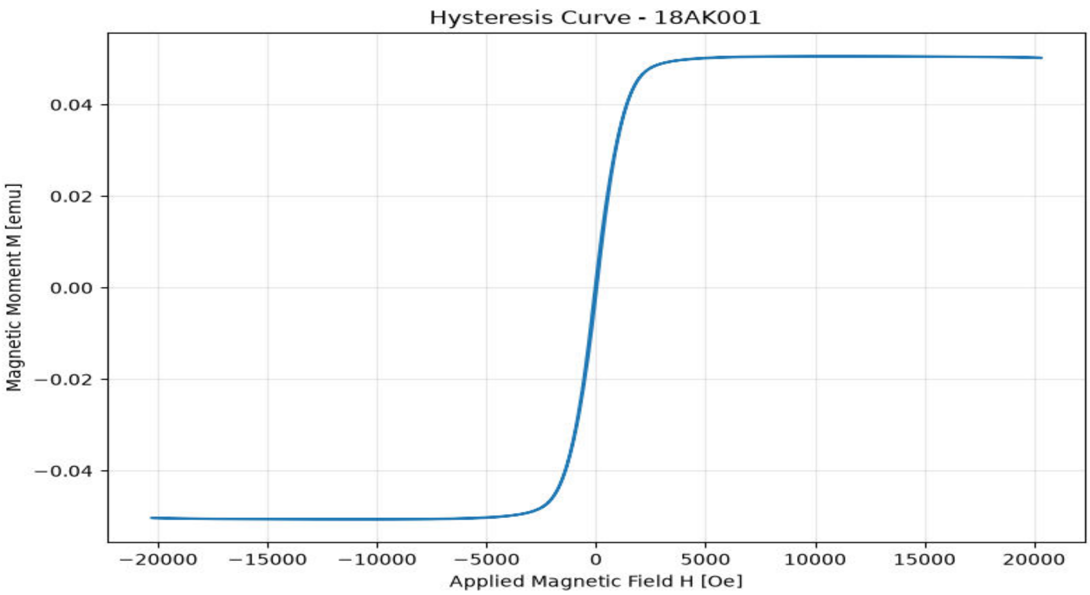


Figure 5: Hysteresis curve of one sample from the dataset

Furthermore, the comparison between the original hysteresis curve and the 400-point resampled curve highlights the impact of data preprocessing on magnetic property analysis. Both curves exhibit the expected hysteresis loop behavior, with steep transitions near zero applied field and saturation at high positive and negative field values.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

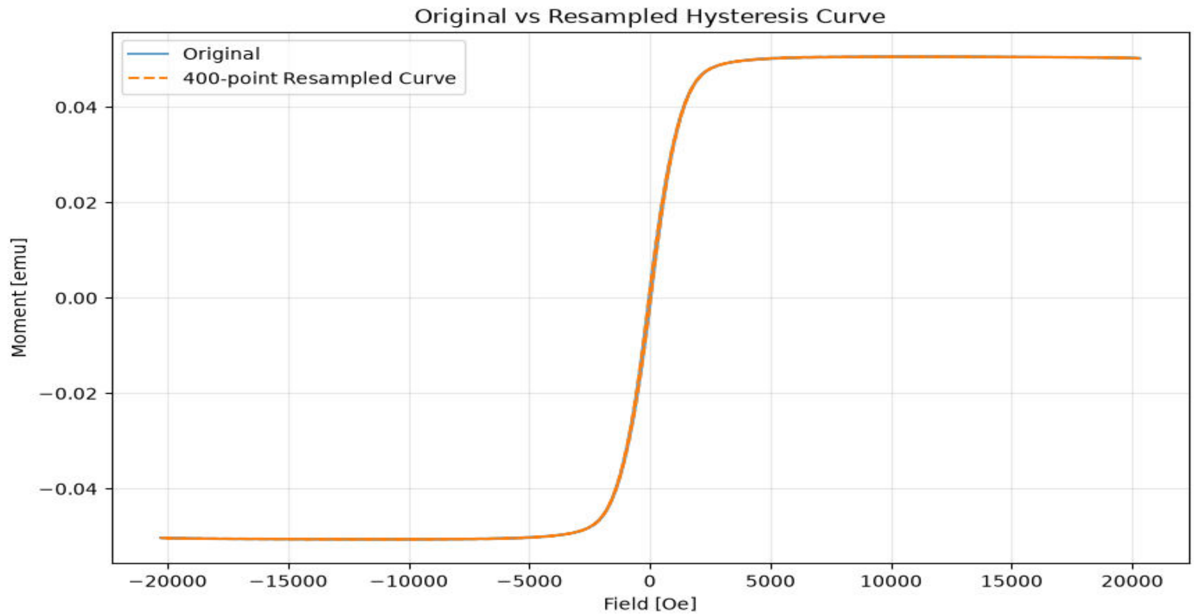


Figure 6: Comparable hysteresis curve of original and resampled data

2.8 Data Training and Testing

The preprocessing workflow structured the hysteresis curve data into arrays suitable for machine learning regression. Each sample was resampled to a fixed number of points (400 points) to ensure uniformity across the dataset, and the resampled field and moment values were concatenated into a single feature vector of length 800. Extracted target magnetic properties such as coercivity, remanence, and saturation magnetization into a separate output array, producing five values per sample. In total, the dataset contained 561 samples, with the feature matrix shaped as (561, 800) and the target matrix shaped as (561, 5). This approach standardized the input length across all samples, optimized the dataset for use in frameworks like Scikit-learn, XGBoost, and LightGBM, and ensured that the hysteresis curve data was clean, consistent, and computationally efficient, providing a robust foundation for the comparative evaluation of Random Forest, Gradient Boosting, XGBoost, LightGBM, and Support Vector Regression models.

The dataset of predictive targets in Figures 7 and 8 provides the numerical values of key magnetic properties extracted from hysteresis curves. These include coercivity, intrinsic coercivity, remanence, saturation moment, and residual induction, which serve as the dependent variables for machine learning regression. The sample entries illustrate the diversity of values across different materials: for example, coercivity ranges from very small values (0.000148) to relatively high values (0.262878), while intrinsic coercivity spans from approximately 41 to 181. Similarly, remanence and saturation moment vary significantly, reflecting differences in magnetic strength and retention among samples. Residual induction values closely track coercivity, highlighting their correlation.

The dataset was divided into training and testing subsets using a standard train-test split procedure. Out of the total 561 samples, 448 were assigned to the training set and 113 to the testing set, corresponding to 80% and 20% of the data, respectively. Each sample contained 800 features, derived from resampled hysteresis curve data, and five target variables, representing magnetic properties such as coercivity, intrinsic coercivity, remanence, saturation moment, and residual induction. A fixed random state was applied to ensure reproducibility of the split, allowing consistent evaluation across different model runs. This partitioning strategy provided a reliable framework for training machine learning regression models while preserving an independent test set for unbiased performance assessment.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

```
X_list = []
y_list = []
sample_ids = []

for sample in complete_samples:

    field_resampled, moment_resampled = resample_curve(
        sample["field"],
        sample["moment"],
        N_POINTS
    )

    features = np.concatenate(
        [
            field_resampled,
            moment_resampled
        ]
    )

    targets = [
        sample["targets"][target]
        for target in TARGET_NAMES
    ]

    X_list.append(features)
    y_list.append(targets)
    sample_ids.append(sample["sample_id"])

X = np.asarray(X_list)
y = np.asarray(y_list)

print("X shape:", X.shape)
print("y shape:", y.shape)
```

```
X shape: (561, 800)
y shape: (561, 5)
```

Snippet 3: Building of machine learning features

```
field_feature_names = [
    f"Field_{i+1}"
    for i in range(N_POINTS)
]

moment_feature_names = [
    f"Moment_{i+1}"
    for i in range(N_POINTS)
]

FEATURE_NAMES = (
    field_feature_names +
    moment_feature_names
)

print("Total features:", len(FEATURE_NAMES))
```

```
Total features: 800
```

Snippet 4: Creation of feature names

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

	Coercivity	Intrinsic coercivity	Remanence	Saturation moment	Residual induction
0	0.021314	41.026972	0.001697	0.050476	0.021325
1	0.001200	181.008605	0.000096	0.000497	0.001200
2	0.000148	49.447800	0.000012	0.000143	0.000148
3	0.199510	61.119097	0.015928	0.283145	0.200153
4	0.262878	68.804106	0.020998	0.282831	0.263872

Figure 7: Data frames of the predictive targets

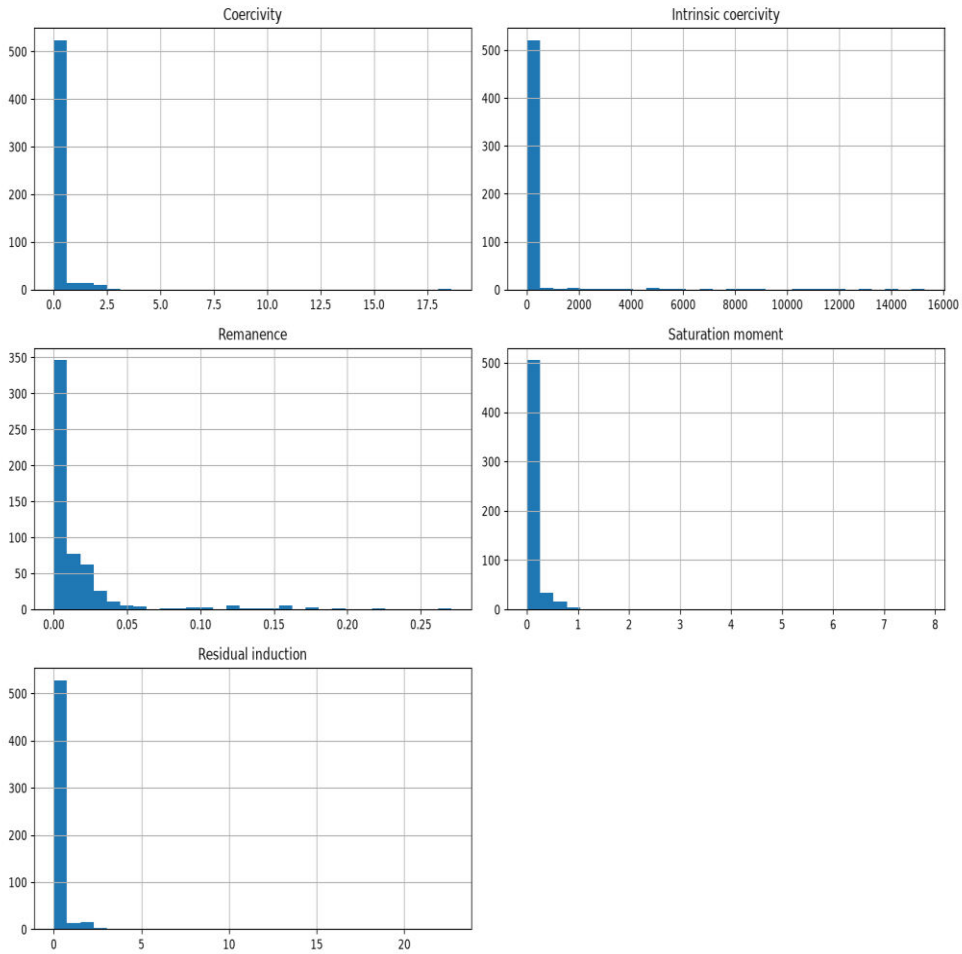


Figure 8: Distribution of predictive targets

The dataset was partitioned into training and testing subsets using a standard train-test split procedure. Out of the total 561 samples, 448 were allocated to the training set and 113 to the testing set, corresponding to 80% and 20% of the data, respectively. Each sample contained 800 features, derived from resampled hysteresis curve data, and five target variables, representing key magnetic properties such as coercivity, intrinsic coercivity, remanence, saturation moment, and residual induction. The use of a fixed random state ensured reproducibility of the split, allowing consistent evaluation across different model runs.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

```
X_train, X_test, y_train, y_test, id_train, id_test = train_test_split(
    X,
    y,
    sample_ids,
    test_size=0.20,
    random_state=RANDOM_STATE
)

print("Training samples:", X_train.shape[0])
print("Testing samples:", X_test.shape[0])
print("Number of features:", X_train.shape[1])
print("Number of targets:", y_train.shape[1])
```

```
Training samples: 448
Testing samples: 113
Number of features: 800
Number of targets: 5
```

Snippet 5: Data splitting of the dataset

1. Random Forest Modeling

The code (Snippet 6) defines and initializes a RandomForestRegressor for predicting magnetic properties from hysteresis curve data. The model was configured with 500 estimators, ensuring a large ensemble of decision trees to improve accuracy and reduce variance. The maximum depth was left unrestricted (None), allowing trees to grow fully and capture complex nonlinear relationships. Parameters such as min_samples_split=2 and min_samples_leaf=1 ensured that trees could split aggressively, while max_features="sqrt" controlled feature selection at each split to balance diversity and performance. A fixed random state was applied for reproducibility, and n_jobs=-1 enabled parallel computation across all available cores, improving training efficiency.

```
random_forest_model = RandomForestRegressor(
    n_estimators=500,
    max_depth=None,
    min_samples_split=2,
    min_samples_leaf=1,
    max_features="sqrt",
    random_state=RANDOM_STATE,
    n_jobs=-1
)

random_forest_model
```

```
▾ RandomForestRegressor
► Parameters
```

Snippet 6: Random forest model

2. Gradient Boosting Modeling

The code (Snippet 7) defines a MultiOutputRegressor using GradientBoostingRegressor as the base estimator, which enables simultaneous prediction of multiple magnetic properties from hysteresis curve data. The Gradient Boosting model was configured with 300 estimators, a learning rate of 0.05, and a maximum depth of 3, balancing predictive accuracy with computational efficiency. The loss function was set to "squared_error", ensuring that the model minimized variance in continuous target predictions. A fixed random state was applied for reproducibility, while n_jobs=-1 allowed parallel computation across all available CPU cores, improving training speed.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

```
gradient_boosting_model = MultiOutputRegressor(
    GradientBoostingRegressor(
        n_estimators=300,
        learning_rate=0.05,
        max_depth=3,
        loss="squared_error",
        random_state=RANDOM_STATE
    ),
    n_jobs=-1
)

gradient_boosting_model
```

```
▸ MultiOutputRegressor
▸ estimator: GradientBoostingRegressor
  ▸ GradientBoostingRegressor
```

Snippet 7: Gradient boosting model

3. XGBoost Modeling

The code (Snippet 8) defines a MultiOutputRegressor with XGBRegressor as the base estimator, enabling simultaneous prediction of multiple magnetic properties from hysteresis curve data. The model was configured with 500 estimators, a learning rate of 0.03, and a maximum depth of 6, allowing it to capture complex nonlinear relationships while maintaining generalization. Additional parameters such as min_child_weight=1, subsample=0.8, and colsample_bytree=0.8 were tuned to control overfitting and improve robustness. The objective function was set to "reg:squarederror", ensuring accurate regression performance. A fixed random state guaranteed reproducibility, while n_jobs=-1 enabled parallel computation across all available cores, enhancing efficiency.

```
xgboost_model = MultiOutputRegressor(
    XGBRegressor(
        n_estimators=500,
        learning_rate=0.03,
        max_depth=6,
        min_child_weight=1,
        subsample=0.8,
        colsample_bytree=0.8,
        objective="reg:squarederror",
        random_state=RANDOM_STATE,
        n_jobs=-1
    ),
    n_jobs=-1
)

xgboost_model
```

```
▸ MultiOutputRegressor
▸ estimator: XGBRegressor
  ▸ XGBRegressor
```

Snippet 8: XGBoost Model

4. LightGBM Modeling

The code (Snippet 9) defines a MultiOutputRegressor using LGBMRegressor as the base estimator, enabling simultaneous prediction of multiple magnetic properties from hysteresis curve data. The model was configured with 500 estimators, a learning rate of 0.03, and 31 leaves, providing a balance between accuracy and computational efficiency.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

The maximum depth was set to -1, allowing unrestricted tree growth, while subsampling (0.8) and column sampling (0.8) were applied to reduce overfitting and improve generalization. A fixed random state ensured reproducibility, and verbosity was suppressed for cleaner output. The use of n_jobs=-1 enabled parallel computation across all available cores, enhancing training speed.

```
lightgbm_model = MultiOutputRegressor(
    LGBMRegressor(
        n_estimators=500,
        learning_rate=0.03,
        num_leaves=31,
        max_depth=-1,
        subsample=0.8,
        colsample_bytree=0.8,
        random_state=RANDOM_STATE,
        verbosity=-1
    ),
    n_jobs=-1
)

lightgbm_model
```

```
▸ MultiOutputRegressor
 ▸ estimator: LGBMRegressor
    ▸ LGBMRegressor
```

Snippet 9: LightGBM model

5. Support Vector Regression Modeling

The code (Snippet 10) constructs a machine learning model that integrates preprocessing and regression for predicting multiple magnetic properties from hysteresis curve data. The model begins with a StandardScaler, which normalizes input features to ensure that the SVR model operates effectively across different scales of field and moment values. Following this, a Support Vector Regression (SVR) model is applied with an RBF kernel, enabling the capture of nonlinear relationships in the data. The SVR was configured with C=100 to control regularization strength, epsilon=0.01 to define the margin of tolerance for prediction errors, and gamma="scale" to adjust kernel influence automatically.

To extend SVR for multi-target prediction, the model was wrapped in a MultiOutputRegressor, allowing simultaneous estimation of coercivity, remanence, saturation moment, and other magnetic properties. The use of n_jobs=-1 enabled parallel computation across all available cores, improving efficiency.

International Journal of Research Publications in Engineering, Technology and Management (IJRPETM)

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

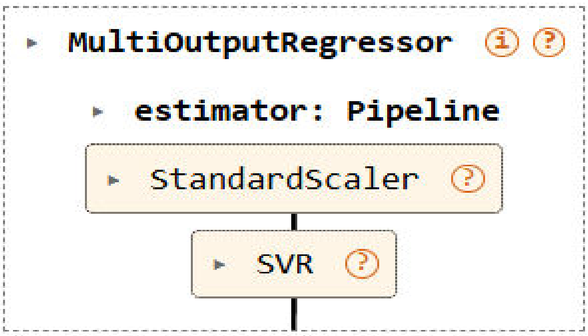
||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

```
svr_base_model = Pipeline(
    [
        (
            "scaler",
            StandardScaler()
        ),
        (
            "svr",
            SVR(
                kernel="rbf",
                C=100,
                epsilon=0.01,
                gamma="scale"
            )
        )
    ]
)

svr_model = MultiOutputRegressor(
    svr_base_model,
    n_jobs=-1
)

svr_model
```



Snippet 10: SVR model

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

```
models = {
    "Random Forest": random_forest_model,
    "Gradient Boosting": gradient_boosting_model,
    "XGBoost": xgboost_model,
    "LightGBM": lightgbm_model,
    "Support Vector Regression": svr_model
}

print(models.keys())
```

Snippet 11: Combination of all models

2.9 Evaluation of the Models

The evaluation function, evaluate_predictions, was used to systematically assess the performance of machine learning models across multiple magnetic property targets. For each target variable, the function computes three key metrics: the coefficient of determination (R^2), which measures how well predictions explain the variance in the data; the mean absolute error (MAE), which quantifies the average magnitude of prediction errors; and the root mean squared error (RMSE), which emphasizes larger deviations by squaring errors before averaging. These metrics are calculated by comparing the true values (y_true) with the predicted values (y_pred) for each target. The results are stored in a structured list of dictionaries, with each entry containing the model name, target variable, and its corresponding R^2, MAE, and RMSE values.

```
def evaluate_predictions(
    model_name,
    y_true,
    y_pred,
    target_names
):

    results = []

    for i, target in enumerate(target_names):

        actual = y_true[:, i]
        predicted = y_pred[:, i]

        r2 = r2_score(
            actual,
            predicted
        )

        mae = mean_absolute_error(
            actual,
            predicted
        )

        rmse = np.sqrt(
            mean_squared_error(
                actual,
                predicted
            )
        )
        results.append(
            {
                "Model": model_name,
                "Target": target,
                "R2": r2,
                "MAE": mae,
                "RMSE": rmse
            }
        )

    return results
```

Snippet 12: Evaluation of the models

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

```
all_results = []
predictions = {}
trained_models = {}

for model_name, model in models.items():

    print("=" * 70)
    print("Training:", model_name)
    print("=" * 70)

    model.fit(
        X_train,
        y_train
    )

    y_pred = model.predict(
        X_test
    )

    predictions[model_name] = y_pred
    trained_models[model_name] = model

    model_results = evaluate_predictions(
        model_name,
        y_test,
        y_pred,
        TARGET_NAMES
    )

    all_results.extend(
        model_results
    )

    print(model_name, "completed.\n")
```

Snippet 13: Training and evaluation of all models

III. RESULTS

3.1 Overall Evaluation of the Models

Table 1 presents the overall evaluation of the models. The Random Forest Regressor demonstrated mixed performance across the five magnetic property prediction targets. For intrinsic coercivity, the model achieved a strong coefficient of determination ($R^2 = 0.913$) with relatively low error values (MAE ≈ 190.73, RMSE ≈ 690.35), indicating that it captured the variance in this property effectively. Similarly, predictions for remanence ($R^2 = 0.689$, MAE ≈ 0.0094, RMSE ≈ 0.0250) and saturation moment ($R^2 = 0.710$, MAE ≈ 0.0327, RMSE ≈ 0.1666) showed moderate accuracy. This reflects the model's ability to generalize nonlinear relationships in hysteresis curve features.

However, the model struggled with coercivity ($R^2 = 0.114$, MAE ≈ 0.256, RMSE ≈ 1.680) and residual induction ($R^2 = 0.088$, MAE ≈ 0.293, RMSE ≈ 2.058), where predictive accuracy was low and errors were relatively high. These results indicate that Random Forest was more effective for properties with stronger nonlinear dependencies (like intrinsic coercivity and remanence) but less reliable for parameters with subtle variations or weaker correlations to the input features.

The Gradient Boosting Regressor showed strong improvements over Random Forest in several targets. For intrinsic coercivity, it achieved an excellent $R^2 = 0.959$ with lower errors (MAE ≈ 119.45, RMSE ≈ 475.90), outperforming Random Forest significantly. Predictions for remanence were highly accurate ($R^2 = 0.906$, MAE ≈ 0.0052, RMSE ≈ 0.0137), while the saturation moment also improved moderately ($R^2 = 0.738$, MAE ≈ 0.0177, RMSE ≈ 0.1583). However, performance on coercivity ($R^2 = 0.242$) and residual induction ($R^2 = 0.195$) remained relatively weak, though slightly better than Random Forest. Overall, Gradient Boosting provided more reliable and consistent predictions, especially for intrinsic coercivity and remanence.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

The XGBoost Regressor demonstrated a similarly competitive performance, particularly excelling in remanence prediction with the highest accuracy among all models ($R^2 = 0.921$, MAE ≈ 0.0044, RMSE ≈ 0.0126). Intrinsic coercivity was also well captured ($R^2 = 0.921$, MAE ≈ 168.70, RMSE ≈ 658.20), comparable to Random Forest but slightly below Gradient Boosting. Predictions for saturation moment ($R^2 = 0.689$) and coercivity ($R^2 = 0.246$) were moderate, while residual induction ($R^2 = 0.207$) remained challenging. XGBoost's strength lies in its robust handling of nonlinear dependencies, making it one of the top-performing models for remanence and intrinsic coercivity.

The LightGBM Regressor showed mixed outcomes. It performed well on remanence ($R^2 = 0.887$, MAE ≈ 0.0053, RMSE ≈ 0.0151), but struggled with intrinsic coercivity ($R^2 = 0.745$, MAE ≈ 415.85, RMSE ≈ 1180.63), where errors were significantly higher than other boosting methods. Predictions for saturation moment ($R^2 = 0.595$) and coercivity ($R^2 = 0.201$) were moderate, while residual induction ($R^2 = 0.166$) remained weak. LightGBM's efficiency and scalability are notable, but in this study, its predictive accuracy was less competitive compared to Gradient Boosting and XGBoost.

	Model	Target	R2	MAE	RMSE
0	Random Forest	Coercivity	0.114467	0.256447	1.680496
1	Random Forest	Intrinsic coercivity	0.912749	190.729288	690.350141
2	Random Forest	Remanence	0.689008	0.009425	0.025049
3	Random Forest	Saturation moment	0.710045	0.032717	0.166614
4	Random Forest	Residual induction	0.087689	0.292710	2.058359
5	Gradient Boosting	Coercivity	0.242160	0.209567	1.554618
6	Gradient Boosting	Intrinsic coercivity	0.958536	119.451204	475.904092
7	Gradient Boosting	Remanence	0.906411	0.005171	0.013741
8	Gradient Boosting	Saturation moment	0.738389	0.017720	0.158261
9	Gradient Boosting	Residual induction	0.195365	0.242829	1.933076
10	XGBoost	Coercivity	0.245542	0.198153	1.551146
11	XGBoost	Intrinsic coercivity	0.920686	168.702211	658.204839
12	XGBoost	Remanence	0.921019	0.004366	0.012623
13	XGBoost	Saturation moment	0.688519	0.033525	0.172688
14	XGBoost	Residual induction	0.207277	0.232685	1.918714
15	LightGBM	Coercivity	0.201104	0.208869	1.596174
16	LightGBM	Intrinsic coercivity	0.744812	415.849943	1180.633742
17	LightGBM	Remanence	0.887490	0.005333	0.015066
18	LightGBM	Saturation moment	0.594638	0.060342	0.197000
19	LightGBM	Residual induction	0.165530	0.244151	1.968589
20	Support Vector Regression	Coercivity	0.106789	0.177829	1.687766
21	Support Vector Regression	Intrinsic coercivity	-0.057247	613.308228	2403.108353
22	Support Vector Regression	Remanence	0.780861	0.007376	0.021027
23	Support Vector Regression	Saturation moment	0.752843	0.026746	0.153827
24	Support Vector Regression	Residual induction	0.079803	0.213504	2.067235

Table 1: Overall evaluation of the models

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

3.2 Comparative Evaluation of the Models

1. R^2 Comparative Evaluation of the Models

The comparative analysis of R^2 values highlights clear differences in predictive accuracy among the five machine learning models. Gradient Boosting consistently achieved the highest scores, excelling in intrinsic coercivity ($R^2 = 0.959$) and remanence ($R^2 = 0.906$), while also performing strongly on saturation moment ($R^2 = 0.738$). XGBoost closely followed, delivering the best performance for remanence ($R^2 = 0.921$) and competitive results for intrinsic coercivity ($R^2 = 0.921$), confirming its robustness in modeling nonlinear dependencies.

Random Forest showed solid results for intrinsic coercivity ($R^2 = 0.913$) but weaker performance in other targets, particularly coercivity ($R^2 = 0.114$) and residual induction ($R^2 = 0.088$). LightGBM demonstrated efficiency but lower accuracy, with moderate results for remanence ($R^2 = 0.887$) and weaker outcomes for intrinsic coercivity ($R^2 = 0.745$) and saturation moment ($R^2 = 0.595$). Support Vector Regression (SVR) provided strong nonlinear modeling for saturation moment ($R^2 = 0.753$) and remanence ($R^2 = 0.781$), but failed on intrinsic coercivity, yielding a negative R^2 (-0.057), indicating poor predictive capability.

Overall, the R^2 evaluation shows that Gradient Boosting and XGBoost are the most reliable models, consistently achieving high explanatory power across multiple targets. SVR performed well for certain nonlinear properties but lacked robustness, while Random Forest and LightGBM offered moderate results with specific strengths and weaknesses.

Target	Coercivity	Intrinsic coercivity	Remanence	Residual induction	Saturation moment
Model					
Gradient Boosting	0.242160	0.958536	0.906411	0.195365	0.738389
LightGBM	0.201104	0.744812	0.887490	0.165530	0.594638
Random Forest	0.114467	0.912749	0.689008	0.087689	0.710045
Support Vector Regression	0.106789	-0.057247	0.780861	0.079803	0.752843
XGBoost	0.245542	0.920686	0.921019	0.207277	0.688519

Table 2: R^2 comparison of the models

	Model	Mean_R2	Rank
0	Gradient Boosting	0.608172	1
1	XGBoost	0.596609	2
2	LightGBM	0.518715	3
3	Random Forest	0.502792	4
4	Support Vector Regression	0.332610	5

Table 3: Average and rank of the models

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

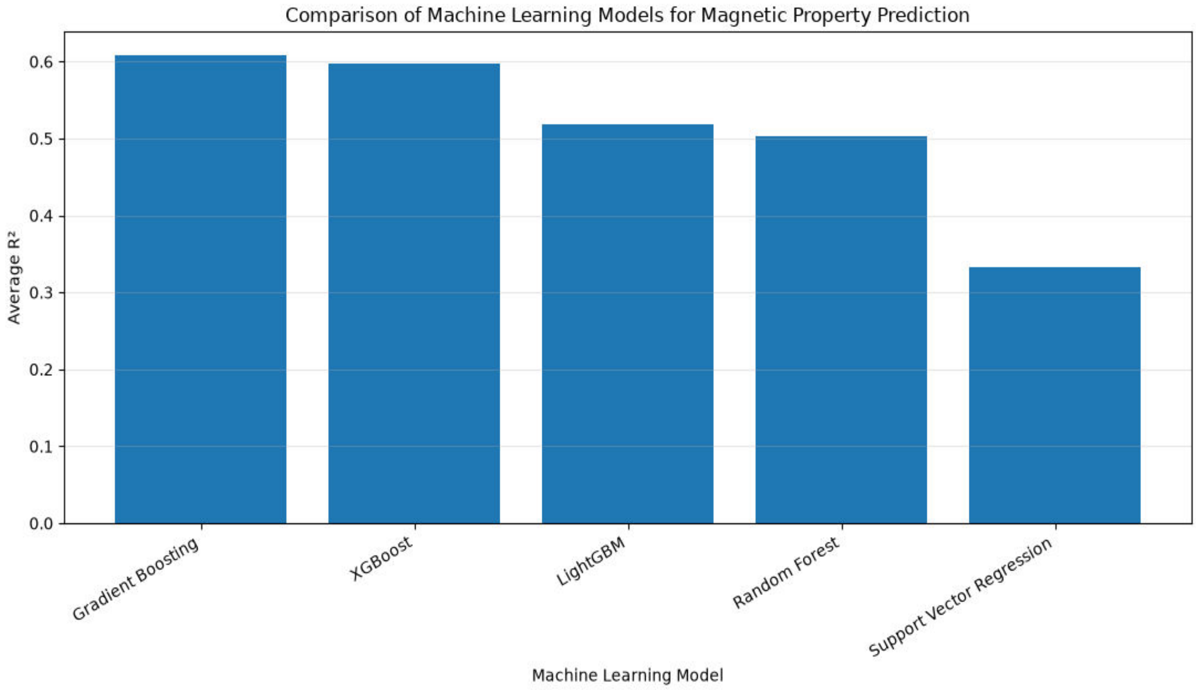


Figure 9: Chart of performance of the comparable models

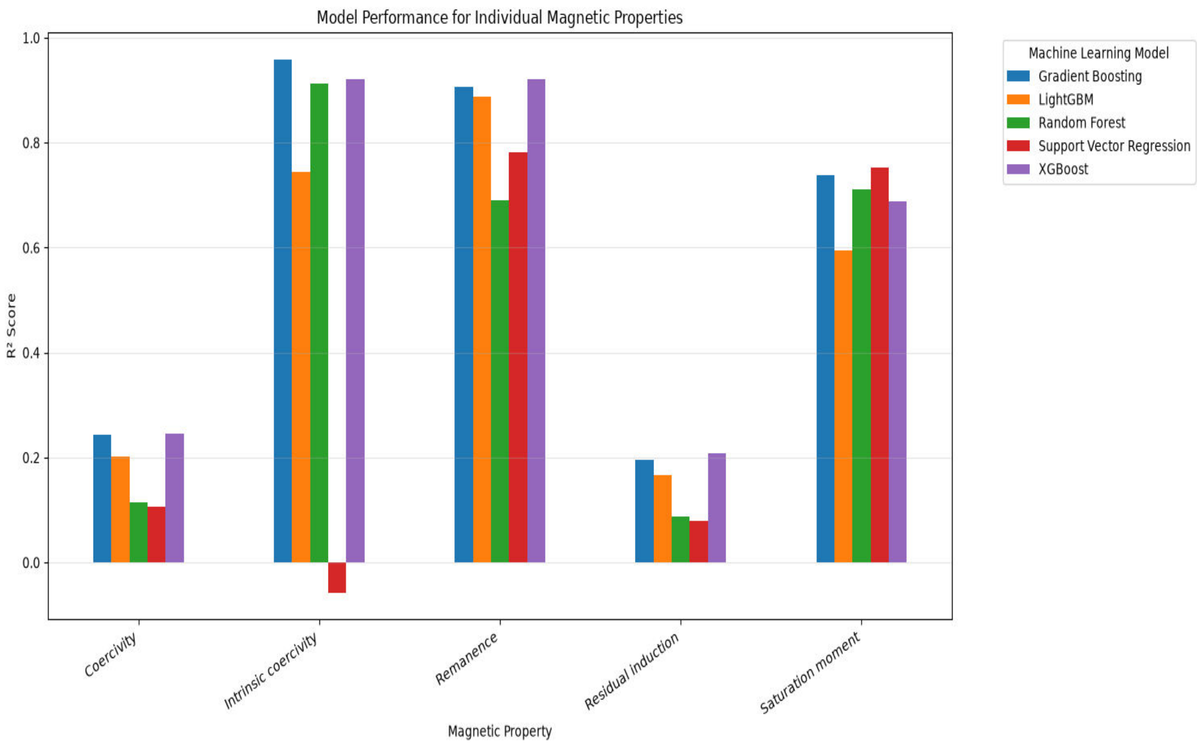


Figure 10: Chart of individual performance of the comparable models

2. RMSE Comparative Evaluation of the Models

The comparison of Root Mean Squared Error (RMSE) values provides insight into the magnitude of prediction errors across models and magnetic property targets. For coercivity, the lowest RMSE was achieved by XGBoost (1.551), closely

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

followed by Gradient Boosting (1.555), while Random Forest (1.680) and SVR (1.688) showed higher error magnitudes. Intrinsic coercivity was best predicted by Gradient Boosting (475.90), significantly outperforming Random Forest (690.35), XGBoost (658.20), LightGBM (1180.63), and SVR (2403.11), where SVR's error was exceptionally large.

For remanence, XGBoost again led with the lowest RMSE (0.0126), followed by Gradient Boosting (0.0137) and LightGBM (0.0151). Random Forest (0.0250) and SVR (0.0210) produced higher errors, indicating weaker precision. In residual induction, XGBoost (1.919) and Gradient Boosting (1.933) performed best, while LightGBM (1.969), Random Forest (2.058), and SVR (2.067) showed larger deviations. For the saturation moment, SVR achieved the lowest RMSE (0.154), slightly outperforming Gradient Boosting (0.158), while Random Forest (0.167), XGBoost (0.173), and LightGBM (0.197) had higher error values.

Overall, the RMSE evaluation highlights that Gradient Boosting and XGBoost consistently minimized errors across most targets, with XGBoost excelling in remanence and coercivity, and Gradient Boosting dominating intrinsic coercivity. SVR showed strength in saturation moment but failed dramatically in intrinsic coercivity. LightGBM, while efficient, generally produced higher error magnitudes compared to the other boosting methods.

Target	Coercivity	Intrinsic coercivity	Remanence	Residual induction	Saturation moment
Model					
Gradient Boosting	1.554618	475.904092	0.013741	1.933076	0.158261
LightGBM	1.596174	1180.633742	0.015066	1.968589	0.197000
Random Forest	1.680496	690.350141	0.025049	2.058359	0.166614
Support Vector Regression	1.687766	2403.108353	0.021027	2.067235	0.153827
XGBoost	1.551146	658.204839	0.012623	1.918714	0.172688

Table 3: RMSE comparison of the models

3. MAE Comparative Evaluation of the Models

The Mean Absolute Error (MAE) comparison highlights the average magnitude of prediction errors across models and magnetic property targets. For coercivity, the lowest MAE was achieved by Support Vector Regression (0.178), followed closely by XGBoost (0.198) and Gradient Boosting (0.210). Random Forest (0.256) produced the highest error, indicating weaker precision for this property.

In intrinsic coercivity, Gradient Boosting again led with the lowest MAE (119.45), outperforming XGBoost (168.70) and Random Forest (190.73). LightGBM (415.85) and SVR (613.31) showed much larger errors, confirming their limitations for this target. For remanence, XGBoost achieved the best MAE (0.0044), followed closely by Gradient Boosting (0.0052) and LightGBM (0.0053). SVR (0.0074) and Random Forest (0.0094) produced higher errors, reflecting weaker accuracy.

In residual induction, SVR performed best (0.214), followed by XGBoost (0.233) and Gradient Boosting (0.243). LightGBM (0.244) and Random Forest (0.293) had higher error magnitudes. For saturation moment, Gradient Boosting achieved the lowest MAE (0.0177), outperforming SVR (0.0267), Random Forest (0.0327), XGBoost (0.0335), and LightGBM (0.0603).

Overall, the MAE evaluation shows that Gradient Boosting and XGBoost consistently minimized average errors across most targets, with Gradient Boosting excelling in intrinsic coercivity and saturation moment, while XGBoost dominated remanence. SVR showed strength in coercivity and residual induction but failed in intrinsic coercivity. Random Forest and LightGBM produced higher errors overall, with LightGBM particularly weak in intrinsic coercivity.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

Target	Coercivity	Intrinsic coercivity	Remanence	Residual induction	Saturation moment
Model					
Gradient Boosting	0.209567	119.451204	0.005171	0.242829	0.017720
LightGBM	0.208869	415.849943	0.005333	0.244151	0.060342
Random Forest	0.256447	190.729288	0.009425	0.292710	0.032717
Support Vector Regression	0.177829	613.308228	0.007376	0.213504	0.026746
XGBoost	0.198153	168.702211	0.004366	0.232685	0.033525

Table 4: MAE comparison of the models

3.3 Best Model for the Predictive Targets

The evaluation of predictive performance across magnetic properties revealed that different models excelled in distinct areas. For coercivity, the best-performing algorithm was XGBoost, which achieved the highest explanatory power with an R^2 of 0.246, alongside relatively low error values (MAE ≈ 0.198, RMSE ≈ 1.551). In contrast, intrinsic coercivity was most accurately predicted by Gradient Boosting, delivering an exceptional R^2 of 0.959 and minimizing errors (MAE ≈ 119.45, RMSE ≈ 475.90), clearly outperforming all other models. For remanence, XGBoost again dominated, achieving the strongest results with $R^2 = 0.921$, MAE ≈ 0.0044, and RMSE ≈ 0.0126, reflecting its precision in modeling this property. The saturation moment was best captured by Support Vector Regression (SVR), which leveraged its kernel-based approach to achieve $R^2 = 0.753$, MAE ≈ 0.0267, and RMSE ≈ 0.154, outperforming ensemble methods in this specific target. Finally, for residual induction, XGBoost provided the most reliable predictions, with $R^2 = 0.207$, MAE ≈ 0.233, and RMSE ≈ 1.919. Overall, the findings highlight that Gradient Boosting and XGBoost consistently delivered superior performance across most targets, while SVR offered specialized strength in saturation moment prediction.

	Magnetic Property	Best Model	R2	MAE	RMSE
0	Coercivity	XGBoost	0.245542	0.198153	1.551146
1	Intrinsic coercivity	Gradient Boosting	0.958536	119.451204	475.904092
2	Remanence	XGBoost	0.921019	0.004366	0.012623
3	Saturation moment	Support Vector Regression	0.752843	0.026746	0.153827
4	Residual induction	XGBoost	0.207277	0.232685	1.918714

Table 5: Best models for the predictive targets

	Model	Mean_R2	Mean_MAE	Mean_RMSE
0	Gradient Boosting	0.608172	23.985298	95.912758
4	XGBoost	0.596609	33.834188	132.372002
1	LightGBM	0.518715	83.273728	236.882114
2	Random Forest	0.502792	38.264118	138.856132
3	Support Vector Regression	0.332610	122.746737	481.407642

Model summary of the machine learning regression

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

IV. DISCUSSIONS

The comparative evaluation of machine learning models for predicting magnetic properties from hysteresis curve data highlights the strengths and limitations of different approaches. Gradient Boosting and XGBoost consistently emerged as the most reliable models, achieving high R^2 values and low error metrics across multiple targets. This aligns with recent literature emphasizing the superiority of boosting-based methods in magnetic property prediction. For instance, Ai [23] demonstrated that XGBoost achieved higher prediction accuracy, lower mean squared error, and R^2 values closer to one when applied to excitation waveform classification and iron loss prediction. Their study further integrated genetic algorithms for dual-objective optimization, which validates the versatility of XGBoost in both predictive modeling and design optimization for high-frequency magnetic components.

Similarly, Hao, et al. [19] reported that XGBoost outperformed other decision tree algorithms in both classification and regression tasks for perovskite materials, achieving an R^2 of 0.919 in predicting magnetic moments. This corroborates our findings, where XGBoost excelled in predicting remanence ($R^2 = 0.921$) and intrinsic coercivity ($R^2 = 0.921$). This confirms its robustness in handling complex nonlinear dependencies. Meanwhile, Gradient Boosting demonstrated exceptional accuracy in intrinsic coercivity ($R^2 = 0.959$), consistent with its reputation for strong generalization when tuned appropriately.

The role of regularization and hybrid ensemble methods is also evident in recent work. Shi and Jin [20] introduced ridge regression regularization and enhanced cascade forests incorporating XGBoost and LightGBM, achieving improved goodness-of-fit and robust core loss prediction. Their use of SHAP values for feature importance analysis highlights the interpretability advantage of boosting methods, which is crucial for understanding the physical significance of hysteresis curve features. In our study, LightGBM showed efficiency but lower accuracy compared to Gradient Boosting and XGBoost, reflecting the trade-off between computational speed and predictive precision.

Interestingly, Support Vector Regression (SVR) excelled in predicting saturation moment ($R^2 = 0.753$), outperforming ensemble methods in this specific target. This suggests that kernel-based approaches may be particularly effective for properties with strong nonlinear relationships but limited variance. However, SVR's poor performance in intrinsic coercivity (negative R^2) highlights its sensitivity to high-dimensional feature spaces and variance-heavy targets.

V. CONCLUSION

This study systematically compared five machine learning models, Random Forest, Gradient Boosting, XGBoost, LightGBM, and Support Vector Regression for predicting magnetic properties from hysteresis curve data. The results demonstrated that Gradient Boosting and XGBoost consistently delivered superior performance, achieving high R^2 values and low error metrics across most targets. Specifically, Gradient Boosting excelled in intrinsic coercivity, while XGBoost dominated in remanence, coercivity, and residual induction. Support Vector Regression showed specialized strength in saturation moment, highlighting the potential of kernel-based methods for nonlinear targets. These findings align with recent literature and preceding studies, which emphasize the robustness of boosting algorithms in magnetic property prediction and their adaptability to optimization and hybrid ensemble strategies.

VI. LIMITATIONS OF THE RESEARCH

1. Dataset size: The dataset contained 561 samples, which, while sufficient for comparative analysis, may limit generalization to broader material classes.
2. Feature representation: Features were derived from resampled hysteresis curves; alternative representations (e.g., frequency-domain features) may capture additional physical insights.
3. Model scope: The study focused on regression models; classification tasks (e.g., magnetic state prediction) were not explored.
4. Generalizability: Results are specific to the studied hysteresis dataset and may not directly extend to other magnetic materials or operating conditions without retraining.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

VII. RECOMMENDATIONS

The following recommendations are outlined for future research and further studies. These recommendations are:
1. Adopt boosting methods: Gradient Boosting and XGBoost should be prioritized for predictive modeling of magnetic properties.
2. Integrate optimization techniques: A combination of predictive models with optimization algorithms (e.g., genetic algorithms) can enhance both accuracy and energy efficiency.
3. Leverage hybrid ensembles: Hybrid frameworks that integrate XGBoost and LightGBM with other ensemble methods can improve robustness and interpretability.
4. Apply interpretability tools: SHAP values or permutation importance should be used to analyze feature contributions, offering physical insights into hysteresis curve behavior.
5. Strengthen methodological rigor: To achieve publication-quality results, future work should incorporate:

- 5-fold cross-validation for robust performance assessment.
- Hyperparameter optimization for fine-tuned accuracy.
- Log transformation for skewed targets.
- Permutation or SHAP feature importance analysis.
- Statistical comparison across algorithms
- External validation on unseen hysteresis measurements

REFERENCES

[1] P. Adigun, T. Oyekanmi, and A. Adeniyi, "Simulation Prediction of Background Radiation Using Machine Learning," ASRJETS, vol. 101, no. 1, pp. 71–96, 02/03 2025. [Online]. Available: https://asrjetsjournal.org/American_Scientific_Journal/article/view/11432.
[2] P. O. Adigun, A. A. Adeniyi, and T. T. Oyekanmi, "Detection and Interpretation of X-Ray Scans for the Presence of Pneumonia Using Convolutional Neural Network," American Academic Scientific Research Journal for Engineering, Technology, and Sciences, Original vol. 101, no. 1, pp. 97–108, 2025. [Online]. Available: https://core.ac.uk/download/pdf/640473726.pdf. Yes.
[3] T. Oyekanmi, P. Adigun, A. Adeniyi, and N. A. Azeez, "Deep Learning-Based Diagnosis of Brain Cancer Using Convolutional Neural Networks On MRI Scans: A Comparative Study of Model Architectures and Tumor Classification Accuracy," American Scientific Research Journal for Engineering, Technology, and Sciences, vol. 103, pp. 147–163, 04/10 2025. [Online]. Available: https://asrjetsjournal.org/American_Scientific_Journal/article/view/12059.
[4] A. A. Adeniyi, T. T. Oyekanmi, V. Kolawole, O., P. O. Adigun, and N. A. A. Azeez, "Applications of Artificial Intelligence Models in Teletherapy: A Review of Efficacy, and Ethical Implications," American Scientific Research Journal for Engineering, Technology, and Sciences, vol. 103, no. 1, pp. 373–391, %11/%26 2025. [Online]. Available: https://asrjetsjournal.org/American_Scientific_Journal/article/view/12139.
[5] T. T. Oyekanmi, P. O. Adigun, and A. A. Adeniyi, "Design and Evaluation of a Convolutional Neural Network Model for Automated Detection of Diabetic Retinopathy Using Retinal Fundus Photographs," American Scientific Research Journal for Engineering, Technology, and Sciences, vol. 103, no. 1, pp. 313–329, %11/%01 2025. [Online]. Available: https://asrjetsjournal.org/American_Scientific_Journal/article/view/12111.
[6] T. E. Awopejo, P. O. Adigun, and N. A. A. Azeez, "Application of Artificial Intelligence and Machine Learning in Seismological Studies," American Scientific Research Journal for Engineering, Technology, and Sciences, vol. 102, no. 1, pp. 20–38, %05/%23 2025. [Online]. Available: https://asrjetsjournal.org/American_Scientific_Journal/article/view/11684.
[7] T. E. Awopejo, N. A. A. Azeez, and P. O. Adigun, "Review on Glaciological Studies Around the World," American Scientific Research Journal for Engineering, Technology, and Sciences, vol. 102, no. 1, pp. 212–226, %06/%22 2025. [Online]. Available: https://asrjetsjournal.org/American_Scientific_Journal/article/view/11756.
[8] T. E. Awopejo, N. A. A. Azeez, and P. O. Adigun, "Impact of Human Activities on Earthquake Occurrence- a Global Seismological Review," American Scientific Research Journal for Engineering, Technology, and Sciences, vol. 102, no. 1, pp. 440–457, %08/%10 2025. [Online]. Available: https://asrjetsjournal.org/American_Scientific_Journal/article/view/11992.
[9] M. Ait Amou, K. Xia, S. Kamhi, and M. Mouhafid, "A Novel MRI Diagnosis Method for Brain Tumor Classification Based on CNN and Bayesian Optimization," Healthcare, vol. 10, no. 3, p. 494, 2022. [Online]. Available: https://www.mdpi.com/2227-9032/10/3/494.
[10] N. Ida, "Magnetic Materials and Properties," in Engineering Electromagnetics. Cham: Springer International Publishing, 2021, pp. 419–503.

|www.ijrpetm.com | ISSN: 2454-7875 | editor@ijrpetm.com |A Bimonthly, Peer Reviewed & Scholarly Journal|

||Volume 8, Issue 6, November-December 2025||

DOI:10.15662/IJRPETM.2025.0806048

[11] S. B. Dalavi, A. B. Patil, and R. N. Panda, "Magnetic Nanoparticles-Based Coated Materials," in Handbook of Materials Science, Volume 2: Magnetic Materials, R. S. Ningthoujam and A. K. Tyagi Eds. Singapore: Springer Nature Singapore, 2024, pp. 533–571.

[12] A. Raja and S. Aich, "Magnetic properties of gadolinium and/or dysprosium substituted Sm–Co nanocomposite ribbons," Journal of Rare Earths, vol. 42, no. 6, pp. 1093–1100, 2024/06/01/ 2024, doi: https://doi.org/10.1016/j.jre.2023.08.003.

[13] S. Malhotra, M. Chitkara, L. Gupta, and M. Parmar, "Introduction to Nanomagnetic Materials for Electronic Devices," in Nanodevices for Integrated Circuit Design, 2023, pp. 243–272.

[14] G. Mörée and M. Leijon, "Review of Hysteresis Models for Magnetic Materials," Energies, vol. 16, no. 9, p. 3908, 2023. [Online]. Available: https://www.mdpi.com/1996-1073/16/9/3908.

[15] J.-X. Li, J.-P. Cheng, C. Zhang, C.-X. Qu, X.-H. Zhang, and W.-Q. Jiang, "Seismic response study of a steel lattice transmission tower considering the hysteresis characteristics of bolt joint slippage," Engineering Structures, vol. 281, p. 115754, 2023/04/15/ 2023, doi: https://doi.org/10.1016/j.engstruct.2023.115754.

[16] Electrical Academia. "Hysteresis Loop Explained." Electrical Academia. https://electricalacademia.com/electromagnetism/hysteresis-loop-magnetization-curve/ (accessed 25th August, 2025, 2025).

[17] K. Gandha, R. P. Chaudhary, M. J. Kramer, R. T. Ott, D. Paudyal, and I. C. Nlebedim, "Microstructural evolutions, phase transformations and hard magnetic properties in polycrystalline Ce–Co–Fe–Cu alloys," Materials Chemistry and Physics, vol. 286, p. 126179, 2022/07/01/ 2022, doi: https://doi.org/10.1016/j.matchemphys.2022.126179.

[18] V. A. Milyutin and N. N. Nikulchenkov, "Machine Learning Application for Functional Properties Prediction in Magnetic Materials," Physics of Metals and Metallography, vol. 125, no. 12, pp. 1351–1366, 2024/12/01 2024, doi: 10.1134/S0031918X24601471.

[19] Y. Hao et al., "Classification of magnetic ground States and prediction of magnetic moments for ABX3 perovskites utilizing machine learning techniques," Journal of Electroceramics, 2025/09/03 2025, doi: 10.1007/s10832-025-00424-x.

[20] H. Shi and Z. Jin, "Multi-Condition Magnetic Core Loss Prediction and Magnetic Component Performance Optimization Based on Improved Deep Forest," IEEE Access, vol. 13, pp. 82261–82277, 2025, doi: 10.1109/ACCESS.2025.3562736.

[21] S. Alipour Bonab, F. Berg, W. Song, A. Colle, and M. Yazdani-Asrami, "Advanced deep-learning model for temporal-dependent prediction of dynamic behavior of AC losses in superconducting propulsion motors for hydrogen-powered cryo-electric aircraft," Communications Engineering, vol. 4, no. 1, p. 221, 2025/12/17 2025, doi: 10.1038/s44172-025-00554-8.

[22] D. Kuhlman, "A Python Book: Beginning Python, Adanced." [Online]. Available: https://www.davekuhlman.org/python_book_01.pdf

[23] Y. Ai, "Machine learning based prediction and optimization of iron loss," p. 7245, 25th May 2025. [Online]. Available: https://urn.fi/URN:NBN:fi-fe2025050838431